Bash Workshop

UVic Web Dev Club

Slides: http://bit.ly/bashworkshop
Github: https://github.com/uvicwebdev/bash-intro

Bash History - Bourne Again Shell

A shell or “command-line interpreter” is the interface you interact with.

The first UNIX shell was called the V6 shell (SH) and was invented by Ken Thompson (of Bell Labs) in 1971.

There are many popular shells like: BASH, CSH, and ZSH (1990).

BASH is the most common and often used as a default. BASH was invented in 1977.

Terminal History

Definition: Technically, A terminal window, also referred to as a terminal emulator, is a text-only window in a graphical user interface (GUI) that emulates a console.

OR A program that runs the shell.

Definition from: Linux Information Project.

Console History

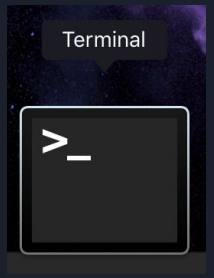
Definition: Is an all-text mode user interface that occupies the entire screen of the display device and which does not sit on top of a GUI.

Prior to the launching of the Apple Macintosh in 1984, virtually all computer displays were consoles.

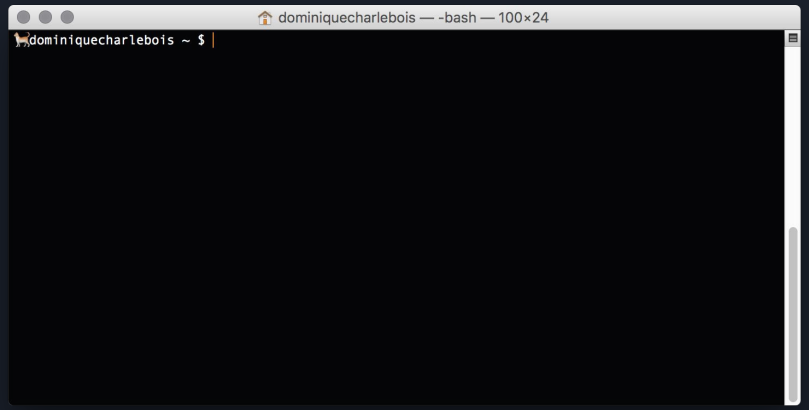
Definition from: Linux Information Project.

Let’s Get Started!

Open your terminal!

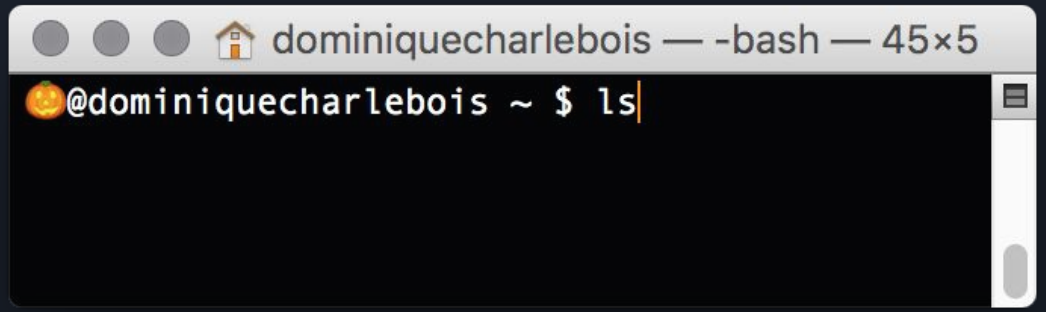


It should look something like this (but not quite)!



Commands

In the command line type “ls”



PRACTICE

- ls
  - list all files and folders in directory
- cd
  - change directory
- mkdir
  - Create a new folder
- touch
  - Create a new file
- cp
  - Copy a file
- mv
  - Move a file or folder

More Available Here:
https://learncodethehardway.org/unix/bash_cheat_sheet.pdf

.bash_profile vs .bashrc

.bash_profile is executed for login shells

.bashrc is executed for interactive non-login shells

Which One?

It is totally up to you. They both serve different purposes.

Note: On OS X the terminal by default runs a login shell every time, so I will be using .bash_profile, but you may wish to use .bashrc

CUSTOMIZE PS1

The default looks something like this: `MacBook:~/Desktop Admin$`

Open either .bash_profile or .bashrc: `open ~/.bash_profile`

- This will open in your default text editor (probably vim), so instead of “open” you can type “nano”.

Type in: `export PS1=" "`

- This will be where we add the customization.

- \d
  - Current date
- \t
  - Current time
- \h
  - Host name
- \#
  - Command number
- \u
  - User Name
- \W
  - Current working directory
- \w
  - Current working directory with full path

In order to see these changes you will need to save the file and “reload” your terminal.

Option 1

- Typing in: `source ~/.bash_profile`

Option 2

- Quite the terminal and re-open

CUSTOMIZE COMMANDS

You can also “shorten” commands by creating alias.

Ex. `alias ga="git add ."`

So now you can type “ga” instead of “git add .”

What is next?

Explore “alternative terminals” such as HyperTerm, iTerm, iTerm2, etc.

Explore “frameworks” such as oh my ZSH!, Bang.sh, etc.

Explore “Package Managers” such as homebrew, macports, etc.

Thank You
UVic Web Dev Club

Join Us Next Week
ECS 104 @ 6pm

References

https://www.ibm.com/developerworks/library/l-linux-shells/index.html
https://programminghistorian.org/lessons/intro-to-bash
http://www.linfo.org/terminal_window.html
http://hawker.me/PimpMyShell-Talk/
https://learncodethehardway.org/unix/bash_cheat_sheet.pdf
https://gist.github.com/stephenll/8762279
https://apple.stackexchange.com/questions/51036/what-is-the-difference-between-bash-profile-and-bashrc
https://learncodethehardway.org/unix/bash_cheat_sheet.pdf